R Notebook

Neil Yetz & Gemma Wallace

Load libraries

```
library(tidyverse)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
## Warning: package 'tibble' was built under R version 3.6.3
## Warning: package 'tidyr' was built under R version 3.6.3
## Warning: package 'dplyr' was built under R version 3.6.3
## Warning: package 'forcats' was built under R version 3.6.3
```

```
library(psych)
library(car)
```

```
## Warning: package 'car' was built under R version 3.6.3
```

```
library(lsr)
library(MBESS)
```

Import Data

```
slp <- read_csv("slpdata.csv")
```

Factor grouping variables

```
slp <- mutate(slp,
 female = ifelse(sex == 1, 0, 1),
 female.f = factor(female, levels = c(0,1), labels = c("male", "female")))

slp <- mutate(slp, cond.f = factor(cond, levels = c(1,2,3),
                                   labels = c("self help", "group-based", "group + partner")))
```

Calculate descriptives

For whole dataset

```
describe(slp)
```

```
##           vars   n   mean     sd median trimmed    mad    min     max   range
```

```
## cond         1 600   2.00   0.82   2.00    2.00   1.48  1.00   3.00   2.00
## prior        2 600   0.72   0.45   1.00    0.78   0.00  0.00   1.00   1.00
## age          3 600  44.94  12.87  45.20   45.12  16.46 20.00  67.80  47.80
## anxiety      4 600   3.88   0.90   3.86    3.89   0.93  1.05   6.84   5.79
## hygiene      5 600   5.99   1.57   6.05    6.04   1.57  1.68   9.74   8.06
## support      6 600   3.04   0.68   2.96    3.02   0.73  1.09   4.91   3.82
## sleep        7 600  68.88  12.14  69.00   69.09  11.86 34.00  99.00  65.00
## lifesat      8 600   4.06   0.92   4.05    4.04   0.96  1.68   6.61   4.93
## sex          9 600   1.41   0.49   1.00    1.39   0.00  1.00   2.00   1.00
## id          10 600 300.50 173.35 300.50  300.50 222.39  1.00 600.00 599.00
## female      11 600   0.41   0.49   0.00    0.39   0.00  0.00   1.00   1.00
## female.f*   12 600   1.41   0.49   1.00    1.39   0.00  1.00   2.00   1.00
## cond.f*     13 600   2.00   0.82   2.00    2.00   1.48  1.00   3.00   2.00
##            skew kurtosis   se
## cond       0.00    -1.50 0.03
## prior     -1.00    -1.01 0.02
## age       -0.10    -1.14 0.53
## anxiety   -0.07    -0.06 0.04
## hygiene   -0.23    -0.29 0.06
## support    0.21    -0.51 0.03
## sleep     -0.16    -0.17 0.50
## lifesat    0.13    -0.23 0.04
## sex        0.36    -1.87 0.02
## id         0.00    -1.21 7.08
## female     0.36    -1.87 0.02
## female.f*  0.36    -1.87 0.02
## cond.f*    0.00    -1.50 0.03
```

Summarize descriptives grouping variables

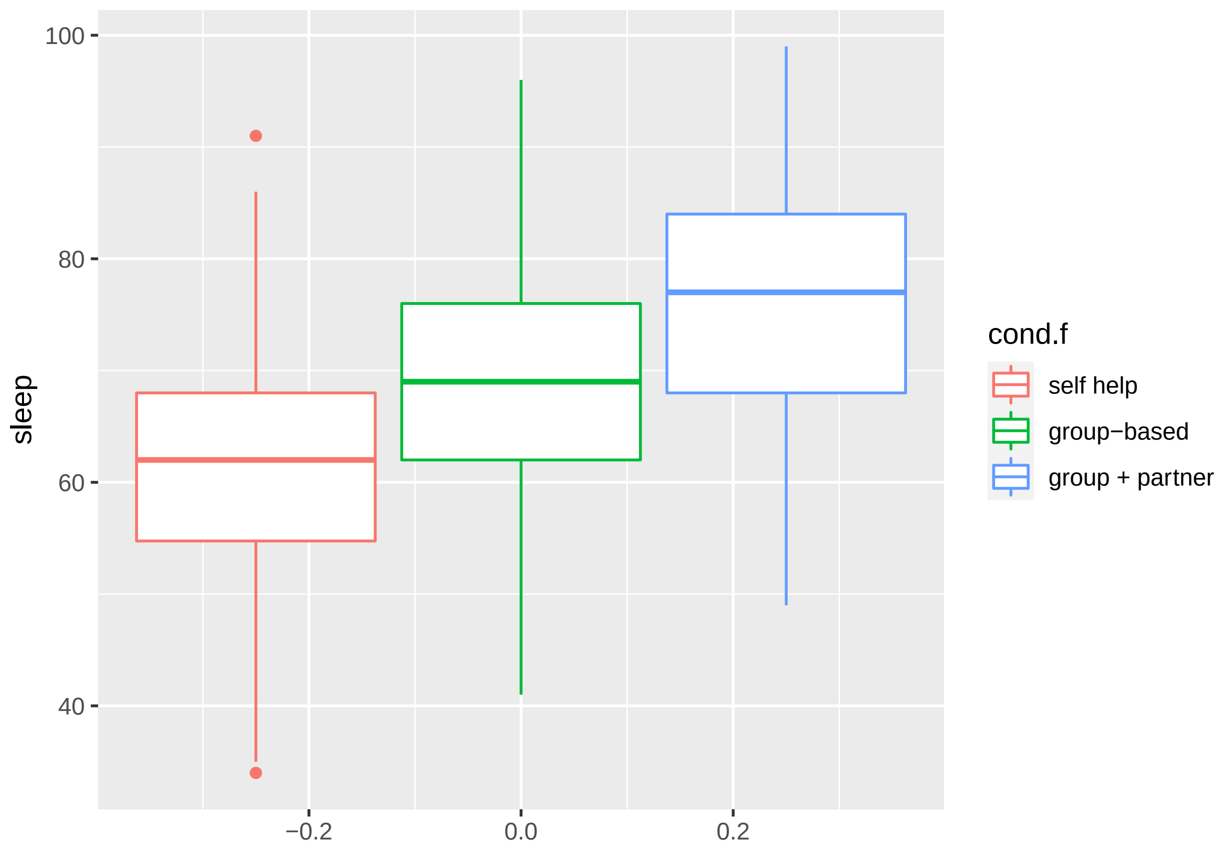
```
aggregate(x=slp$sleep, by=list(slp$female.f, slp$cond.f), FUN=mean)
```

```
##   Group.1           Group.2        x
## 1    male         self help 54.86792
## 2  female         self help 68.34043
## 3    male       group-based 65.01538
## 4  female       group-based 76.70000
## 5    male group + partner 72.75214
## 6  female group + partner 81.36145
```

Visualize the data

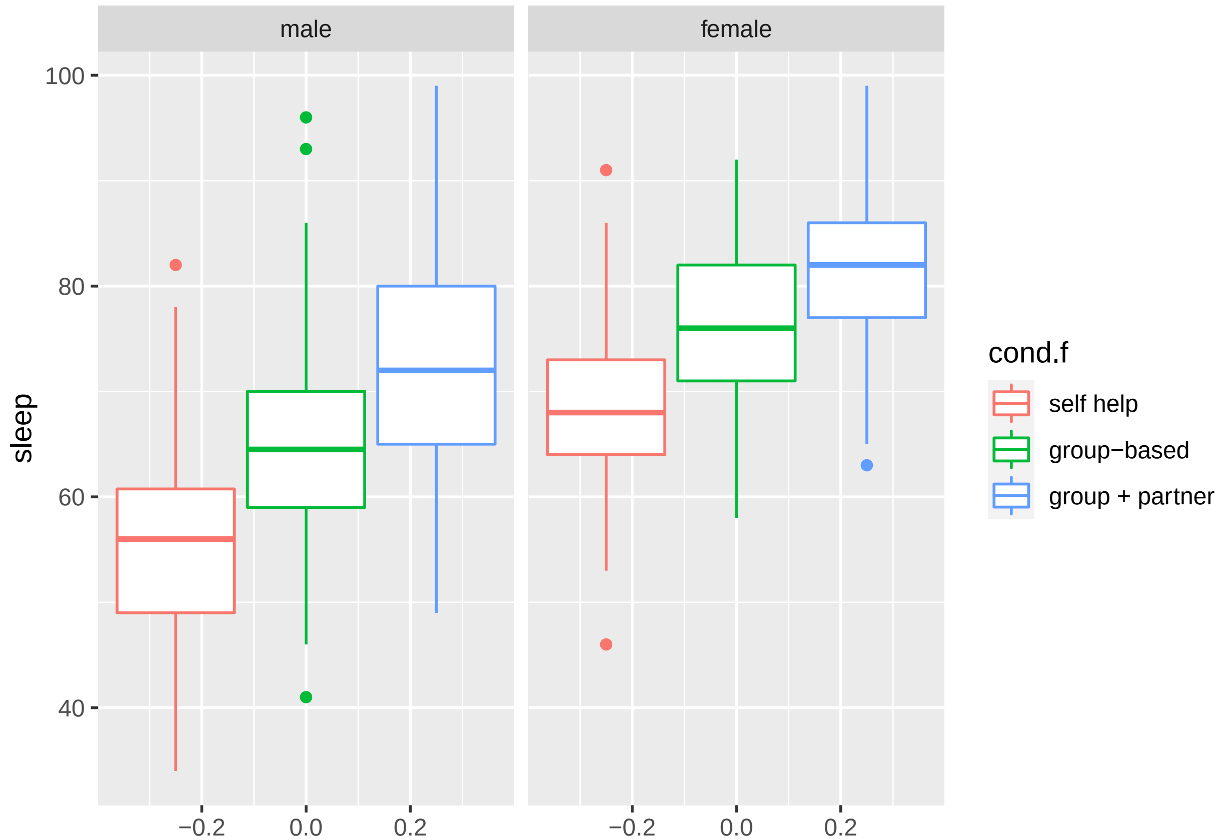
Create boxplots of sleep efficiency across treatment groups

```
ggplot(slp, aes(y = sleep, color = cond.f)) +
  geom_boxplot()
```



Create boxplots of sleep efficiency across treatment groups and sex

```
ggplot(slp, aes(y = sleep, color = cond.f)) +
  geom_boxplot() +
  facet_wrap(~female.f)
```



Conduct a factorial ANOVA

```
model <- lm(sleep ~ female.f + cond.f + female.f*cond.f, data = slp)
Anova(model, type = 3)
```

```
## Anova Table (Type III tests)
##
## Response: sleep
##                 Sum Sq  Df   F value  Pr(>F)
## (Intercept)     319112   1 4073.6899 < 2e-16 ***
## female.f          9043   1  115.4370 < 2e-16 ***
## cond.f           17836   2  113.8443 < 2e-16 ***
## female.f:cond.f    593   2    3.7865 0.02322 *
## Residuals        46531 594
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Calculate effect sizes

```
etaSquared(model, type = 3, anova = FALSE)
```

```
##                      eta.sq eta.sq.part
## female.f        0.102436146  0.16271641
## cond.f          0.202045480  0.27709833
## female.f:cond.f 0.006720116  0.01258869
```

```r
ci.pvaf(F.value = 115.4370, df.1 =1, df.2 = 594, N = 600, conf.level = .95)
```

```
## $Lower.Limit.Proportion.of.Variance.Accounted.for
## [1] 0.1117234
##
## $Probability.Less.Lower.Limit
## [1] 0.025
##
## $Upper.Limit.Proportion.of.Variance.Accounted.for
## [1] 0.2143099
##
## $Probability.Greater.Upper.Limit
## [1] 0.025
##
## $Actual.Coverage
## [1] 0.95
```

```r
ci.pvaf(F.value = 113.8443, df.1 =2, df.2 = 594, N = 600, conf.level = .95)
```

```
## $Lower.Limit.Proportion.of.Variance.Accounted.for
## [1] 0.2173461
##
## $Probability.Less.Lower.Limit
## [1] 0.025
##
## $Upper.Limit.Proportion.of.Variance.Accounted.for
## [1] 0.330051
##
## $Probability.Greater.Upper.Limit
## [1] 0.025
##
## $Actual.Coverage
## [1] 0.95
```

```r
ci.pvaf(F.value = 3.7865,   df.1 =2, df.2 = 594, N = 600, conf.level = .95)
```

```
## $Lower.Limit.Proportion.of.Variance.Accounted.for
## [1] 6.757303e-05
##
## $Probability.Less.Lower.Limit
## [1] 0.025
##
## $Upper.Limit.Proportion.of.Variance.Accounted.for
## [1] 0.03371934
##
## $Probability.Greater.Upper.Limit
## [1] 0.025
##
## $Actual.Coverage
## [1] 0.95
```